

Integrated Querying of Images by Color, Shape, and Texture Content of Salient Objects*

Ediz Şaykol, Uğur GÜDÜKBAY, and Özgür ULUSOY

Department of Computer Engineering, Bilkent University,
06800 Bilkent, Ankara, Turkey
{ediz, gudukbay, oulusoy}@cs.bilkent.edu.tr

Abstract. The growing prevalence of multimedia systems is bringing the need for efficient techniques for storing and retrieving images into and from a database. In order to satisfy the information need of the users, it is of vital importance to effectively and efficiently adapt the retrieval process to each user. Considering this fact, an application for querying the images via their color, shape, and texture features in order to retrieve the similar salient objects is proposed. The features employed in content-based retrieval are most often simple low-level representations, while a human observer judges similarity between images based on high-level semantic properties. Using color, shape, and texture as an example, we show that a more accurate description of the underlying distribution of low-level features improves the retrieval quality. The performance experiments show that our application is effective in retrieval quality and has low processing cost.

1 Introduction

Over the last few years, retrieving images from large collections using image content has been a very important topic. The need to find a desired image from a collection is shared by many professional groups, including medicine, graphic design, criminology, publishing, etc. A considerable amount of information exists in images. Thus, the quick visual access to the stored images would be very advantageous for efficient navigation through image collections. That's why the problems of image retrieval have become widely recognized, and the search for solutions has become an important research area.

Content-based image retrieval systems provide a facility to retrieve the suitable images from the image database for a given query (e.g., [1]). A very common way for query specification is query-by-example. Returning a relevance-ordered list of database images for the query is generally accepted by the researchers. The basic need is to embed a dissimilarity metric among the objects, hence to

* This work is supported in part by Turkish State Planning Organization (DPT) under grant number 2004K120720, and European Commission 6th Framework Program MUSCLE Network of Excellence Project with grant number FP6-507752.

differentiate them, and rank them. Various dissimilarity metrics have been used in the literature (e.g., histogram intersection [2]). These metrics generally rely on feature vectors extracted from the visual content of the images. The feature vectors generally store information based on the color, shape, and texture contents.

In this paper, we present an application that provides an integrated mechanism to query images by color, shape, and texture content of the salient objects. Querying salient objects rather than whole images is more interesting, since users may want to focus on some specific parts of the images for querying purposes. By this type of querying, multiple object regions can be queried on a single image. Video frames can also be processed by our application after the salient objects in frames are extracted by an object extraction tool [3]. The object extraction mechanism is an improved version of the one presented in [4]. The object regions can be determined by simple mouse clicks, polygon specifications, or bounding rectangle drawings. The extracted features (color, shape, and texture vectors) are stored in an object-feature database. In our application, the color vector is based on color histograms but the pixels are probabilistically distance-weighted during computations. Shape vector is a combination of two vectors: the first one is the angular distribution of the pixels around the centroid of the salient object. The second vector is the accumulation of the pixels in the concentric circles centered at the centroid of the object. This type of shape description is very close to the human visual system [5]. Our texture vector is based on Gabor filter banks, which are widely employed in face recognition, vehicle recognition, defect detection, automatic speech recognition, fingerprint matching, etc.

The paper is organized as follows: Section 2 presents the literature summary on low-level object features. We explain our design principles for integrated querying application in Section 3. The performance experiments are discussed in Section 4, and Section 5 concludes the paper.

2 Low-Level Features of Salient Objects

Color, shape, and texture are known as the low-level features in content-based querying terminology. The low-level feature content of the images, or salient objects residing in images, are generally encoded in feature vectors. The feature vectors that we have employed are invariant under scale, rotation, and translation.

2.1 Color

Image data represent physical quantities such as *chromaticity* and *luminance*. Chromaticity is the color quality of light defined by its wavelength. Luminance is the amount of light. To the viewer (i.e., human), these physical quantities are perceived by such attributes as color and brightness [5]. It is the HSI (Hue-Saturation-Intensity) model that represents the colors in a very close way to the human color perception. It is an intuitive representation since it corresponds to

how a painter mixes colors on a palette. Image processing applications – such as histogram operations, intensity transformations, and convolutions – operate only on an image’s intensity. These operations are performed much easier on an image in the HSI color space.

Traditional color histograms [1] generally suffer from low retrieval effectiveness. The retrieval effectiveness can be improved by taking the spatial distribution of the colors into consideration. Color correlograms [6] store the spatial correlation of color regions as well as the global distribution of local spatial correlation of colors in a tabular structure. Although its retrieval effectiveness is better than the traditional color histograms, it is computationally very expensive. In [7], the contributions of the pixels to the color histogram are weighted according to the *Laplacian*, *probabilistic*, and *fuzzy* measures. The weighting is related to a local measure of color non-uniformity (or color activity) determined by some neighborhood of pixels. This weighting approach, without changing the sizes of the histograms, is found to be more effective and its computational complexity is not excessive.

2.2 Shape

The human vision system identifies objects with the edges they contain, both on the boundary and in the interior based on the intensity differences among pixels [5]. These intensity differences are captured as the shape content of salient objects with respect to their centroids in images.

The shape descriptors are classified in two groups: *contour-based* (e.g., Turning Angle representation [8] and Fourier descriptors [9]) and *region-based* (e.g., moment descriptors [10], generic Fourier descriptors [11], and grid descriptors [12]). The former type of descriptors only deals with the object’s boundary and has some major limitations. The latter type of descriptors deals with both the interior and the boundary of the objects, thus can be employed in more general shape retrieval applications. In [11], generic Fourier descriptors are shown to be a more effective shape descriptor than the other contour-based and region-based descriptors.

2.3 Texture

Texture is an important feature since the images can be considered as the composition of different texture regions. There are various techniques for texture feature extraction. The statistical approaches make use of the intensity values of each pixel in an image, and apply various statistical formulae to the pixels in order to calculate feature descriptors [13]. Some systems employ three Tamura [14] texture measures, *coarseness*, *contrast*, and *orientation* (e.g., QBIC system [1]). Photobook [15] exploits texture content based on Wold decomposition; the texture components are *periodicity*, *directionality*, and *randomness*.

Manjunath and Ma [16] have shown that Gabor filter banks are very effective for texture retrieval. They are widely-used in various areas, such as face recognition, vehicle recognition, fingerprint matching, etc. Their use is mainly as follows: texture features are found by calculating the mean and variation of

the Gabor filtered image, processed within a bandpass filter. After applying normalization by a circular shift of the feature elements, the images are re-oriented according to the dominant direction. Then, these feature elements are used to encode the texture content of the image. A comparison of Gabor-filter based texture features can be found in [17].

3 Integrated Querying of Salient Objects

In this paper, we present an application that provides an integrated mechanism to query images by color, shape, and texture content of the salient objects. Querying salient objects rather than whole images is more interesting, since users may want to focus on some specific parts of the images. Moreover, in some application domains, the classification of the salient objects is crucial (e.g., surveillance). Hence, the visual content of the salient objects can be employed for classification.

In our application, we have employed a semi-automatic object extraction scheme to extract object regions. The object extraction scheme is an improved version of the one presented in [4]. The object regions can be determined by simple mouse clicks, polygon specifications, or bounding rectangle drawings. The extracted features (color, shape, and texture vectors) for salient objects are stored in an object-feature database. Video frames can also be processed by our application once the salient objects are extracted by our object extraction tool [3].

In our application, the color vector is based on color histograms but the pixels are probabilistically distance-weighted during computations. Shape vector is a combination of two vectors: the first one is the angular distribution of the pixels around the centroid of the salient object. The second vector is the accumulation of the pixels in the concentric circles centered at the centroid of the object. Our texture vector is based on Gabor filters.

3.1 Color Vector

We have used HSI color space in the computation of the color vector. The main reasons are its perceptual uniformity and its similarity to the human vision system principles. A circular quantization of 20° steps sufficiently separates the hues such that the six primary colors are represented with three sub-divisions. Saturation and intensity are quantized to 3 levels leading to a proper perceptual tolerance along these dimensions. Hence, 18 hues, 3 saturations, 3 intensities, and 4 gray levels are encoded yielding 166 unique colors, i.e., a color feature vector of 166 dimensions (see [18] for details). This color quantization also reduces the effects of noise on the images.

Traditional color histogram technique accumulates pixel intensities into an array of uniquely available intensity values. Since this technique is shown to be not efficient enough in retrieval of similar objects, various techniques have been developed to increase the retrieval effectiveness. We employ a probabilistic distance weighting-scheme similar to [7] for the extracted color vector of salient

objects. The effects of the closer pixels are increased in pixel weight calculation among the neighborhood. In this approach, the contribution of uniform regions is decreased and that of singular points is increased according to the following formula: $w_p = \frac{1}{N_p(c)}$, where w_p is the weight of pixel p , and $N_p(c)$ is the number of pixels having color c as p within the neighborhood of p .

3.2 Shape Vector

The human vision system identifies objects with the edges they contain, both on the boundary and in the interior based on the intensity differences among pixels [5]. These intensity differences, i.e., the shape content, can be captured with respect to the center of mass of these pixels. For this purpose, two specialized feature vectors can be used to encode the shape content of the objects: *distance vector* and *angle vector*.

Distance Vector stores the Euclidean distance between the centroid (c_m) and all of the pixels within the salient object. The distance between a pixel p_i and c_m is re-scaled with respect to the maximum distance among the pixels (i.e., the distance of the farthest pair of pixels). This type of scaled-distance storage in distance vector satisfies scale invariance directly.

Angle Vector stores the counter-clockwise angle between pixel vectors and the unit vector on the x -axis (e_x). The pixel vector v_{p_i} for a pixel p_i is a vector directed from c_m to p_i . The unit vector e_x is translated to c_m . α_{p_i} is the polar angle for p_i . This type of information storage provides an easy and intuitive way to capture angular distribution of the pixels around a fixed object point (c_m).

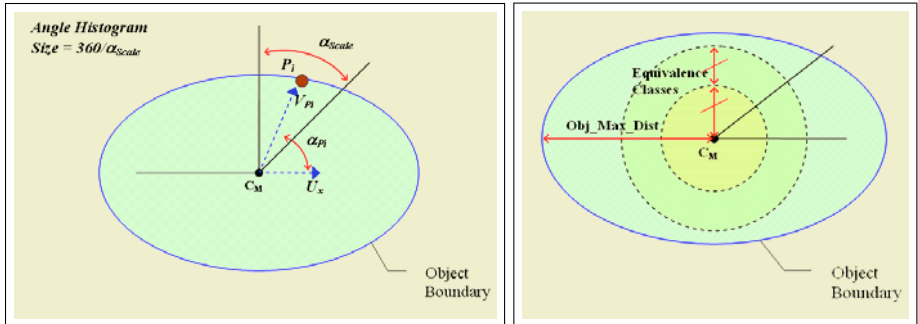


Fig. 1. Visualization of angle and distance vector computations

These two shape vectors are illustrated in Figure 1. The dimension of the angle vector is $\frac{360}{\alpha_{scale}}$. In the experiments, α_{scale} is set to 5. The dimension of the distance vector is fixed at 10, and the radial increments are determined by dynamically-computed **ObjMaxDist** parameter.

3.3 Texture Vector

We have employed a Gabor-filter based texture vector for the texture content of the salient objects. It is shown in [16] that Gabor filter banks are very effective in texture retrieval, and outperform most of the other methods in the literature. Gabor filters are composed of wavelets. Wavelets are a class of functions used to localize a given function in both space and scaling. Wavelets are especially useful for compressing image data. Each wavelet in a Gabor filter holds the energy at a specific frequency and a specific direction. Gabor filters estimate the strength of certain frequency bands and orientations at each location in the image. This gives a result in the spatial domain. The spatial distribution of edges is useful for image querying when the underlying texture is not homogeneous. Processing through a bank of these Gabor filters is approximately equivalent to extracting line edges and bars in the images, at different scales and orientations. Then, the mean and standard deviation of the filtered outputs can be used as features. The details of the Gabor texture feature extraction can be found in [16].

We have selected 5 levels of scales and 6 levels of orientations for the texture feature vector. For each scale and orientation pair, the mean and the standard deviation are calculated. Hence, the dimension of our texture feature vector is 60.

4 Performance Experiments

We have conducted performance experiments on an image library of 100 Brodatz textures [19], 48 carpet patterns gathered from various urls, and 1490 Corel images [20]. The image library contains rotated and scaled versions of the images to validate the invariants of the feature vectors. The experiments were conducted on a 1800 MHz PC with 512 MB RAM.

The retrieval effectiveness is generally evaluated by two well-known metrics, *precision* and *recall*. Precision is the ratio of the number of retrieved images that are relevant to the number of retrieved images. Recall is the ratio of the number of retrieved images that are relevant to the total number of relevant images [21]. The relevance degrees (0, 0.5, 1) are assigned to the extracted objects prior to the experiments to signify the relevance of an object to the query object for each query. Since objects are from various domains, also assigning partial relevance (0.5) to some of the extracted objects is meaningful during retrieval analysis. These relevance degrees are used to calculate precision and recall for each randomly selected query object.

The overall similarity of a query object and a library object is obtained by linearly combining the partial feature vector similarities. Feature vector similarities are computed by histogram intersection method [2]. The weights assigned during the linear combination are set by the user through the interface. In the experiments, we have assigned the following weights: color vector (0.3), angle vector (0.18), distance vector (0.12), and texture vector (0.4). Figure 2 and 3 show two sample query executions corresponding to a scaled Corel image and a carpet pattern, respectively.

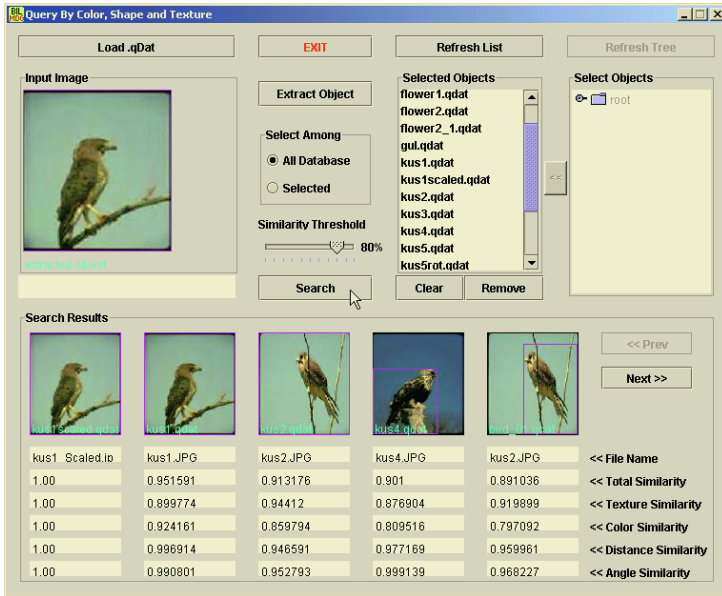


Fig. 2. Sample query 1, a Corel image. The retrieved objects are shown five-by-five

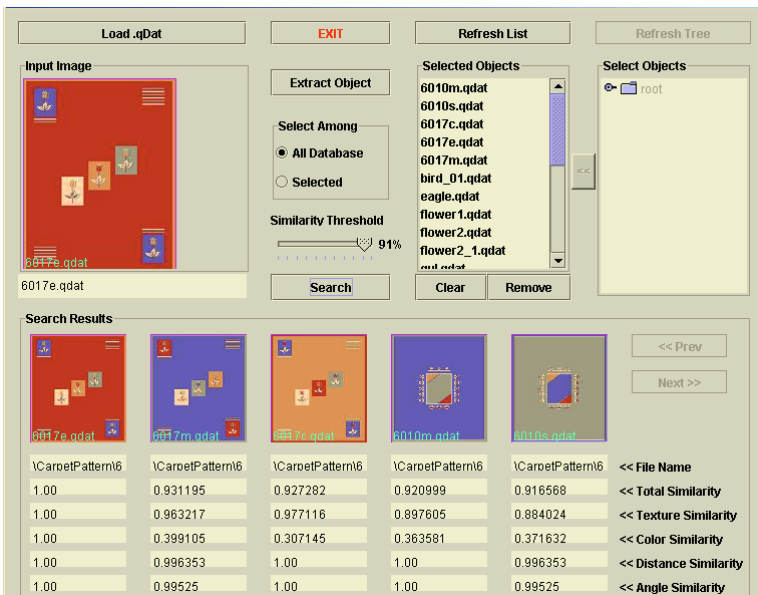


Fig. 3. Sample query 2, a carpet pattern. The retrieved objects are shown five-by-five

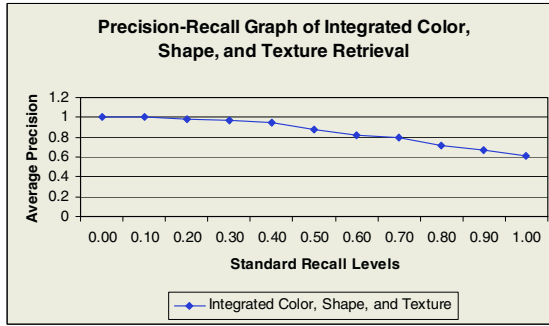


Fig. 4. Precision-recall analysis for integrated querying by color, shape, and texture of salient objects

The precision-recall analysis is carried out by randomly selected 100 query objects. Library objects having similarity greater than 0.80 for a query object are retrieved and their corresponding relevance degrees are used in calculations. The effectiveness is evaluated as the average of the results calculated for each query separately. The individual precision values are interpolated to a set of 11 standard recall levels (0, 0.1, 0.2, ..., 1) to ease the computation of average precision and recall values [21]. Figure 4 presents the precision-recall analysis of our integrated querying application. The results show that our querying methods yields promising results in terms of retrieval effectiveness. The average precision value is above 80% for the recall levels 0 to 0.7. Moreover, the lowest average precision value is 0.61, which seems reasonable for our image library.

5 Conclusion

In this paper, we present an application that provides an integrated mechanism to query images by color, shape, and texture content of the salient objects. Video frames can also be input to the object extraction component of our application. The extracted features (color, shape, and texture vectors) are stored in an object-feature database. Color vector is a variant of color histograms where the object pixels are probabilistically weighted by distance. Shape vector is a combination of angle and distance vectors, which is similar to human visual system. Our texture vector is based on well-known Gabor filters.

We have created an Image library of texture patterns and ordinary images from various domains. The performance experiments indicate that our integrated querying by color, shape, and texture features gives promising results. Hence, our application can be used in many application areas requiring object analysis based on the dissimilarities among the objects.

Acknowledgement. We are grateful to K. Oral Cansızlar, S. Gönül Kızılelma, and Ö. Nurcan Subakan, who worked in the implementation of the system.

References

1. Flickner, M.: Query by image content: the QBIC system. In: IEEE Computer Magazine. Volume 28. (1995) 23–32
2. Swain, M., Ballard, D.: Color indexing. *Int. J. of Computer Vision* **7** (1991) 11–32
3. Dönderler, M., Şaykol, E., Ulusoy, Ö., Gündükbay, U.: BilVideo: A video database management system. *IEEE Multimedia* **10** (2003) 66–70
4. Şaykol, E., Gündükbay, U., Ulusoy, Ö.: A semi-automatic object extraction tool for querying in multimedia databases. In Adali, S., Tripathi, S., eds.: 7th Workshop on Multimedia Information Systems MIS'01. (2001) 11–20
5. Buser, P., Imbert, M.: Vision. MIT Press, Cambridge, Massachusetts (1992)
6. Huang, J., Kumar, S., Mitra, M., Zhu, W., Zabih, R.: Image indexing using color correlograms. In: Proc. of IEEE Conf. on Com. Vis. and Pat. Rec. (1997) 762–768
7. Boujemaa, N., Vertan, C.: Integrated color texture signature for image retrieval. In: Proc. of Int. Conf. on Image and Signal Processing. (2001) 404–411
8. Arkin, E., Chew, P., Huttenlocher, D., Kedem, K., Mitchel, J.: An efficiently computable metric for comparing polygonal shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **13** (1991) 209–215
9. Zahn, C., Roskies, R.: Fourier descriptors for plane closed curves. *IEEE Trans. on Computer* **C-21** (1972) 269–281
10. Kim, H., Kim, J.: Region-based shape descriptor invariant to rotation, scale and translation. *Signal Processing: Image Communication* **16** (2000) 87–93
11. Zhang, D., Lu, G.: Shape based image retrieval using generic fourier descriptors. *Signal Processing: Image Communication* **17** (2002) 825–848
12. Lu, G., Sajjanhar, A.: Region-based shape representation and similarity measure suitable for content-based image retrieval. *Multimedia Systems* **7** (1999) 165–174
13. Haralick, R., Shanmugam, K., Dinstein, I.: Textural features for image classification. *IEEE Trans. on Systems, Man and Cybernetics* **3** (1973) 610–621
14. Tamura, H., Mori, S.: Textural features corresponding to visual perception. *IEEE Trans. on Systems, Man, and Cybernetics* **8** (1978)
15. Pentland, A., Picard, R., Scarloff, S.: Photobook: Tools for content-based manipulation of image databases. In: Proc. of Storage and Retrieval for Image and Video Databases II, SPIE. Volume 2. (1994) 34–47
16. Manjunath, B., Ma, W.: Texture features for browsing and retrieval of image data. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **18** (1996) 837–842
17. Grigorescu, S., Petkov, N., Kruizinga, P.: Comparison of texture features based on gabor filters. *IEEE Trans. on Image Processing* **11** (2002)
18. Smith, J., Chang, S.: Tools and techniques for color image retrieval. In: Proc. of Sto. and Retr. for Im. and Vid. Databases IV. Volume 2670. (1996) 426–437
19. Brodatz, P.: Textures—A Photographic Album for Artists and Designers. Dover Publications, New York (1966)
20. Corel: Image Library, University of California, Berkeley.
<http://elib.cs.berkeley.edu/photos/corel/> (accessed in 2003)
21. K.S.Jones: Information Retrieval Experiment. Butterworth and Co. (1981)